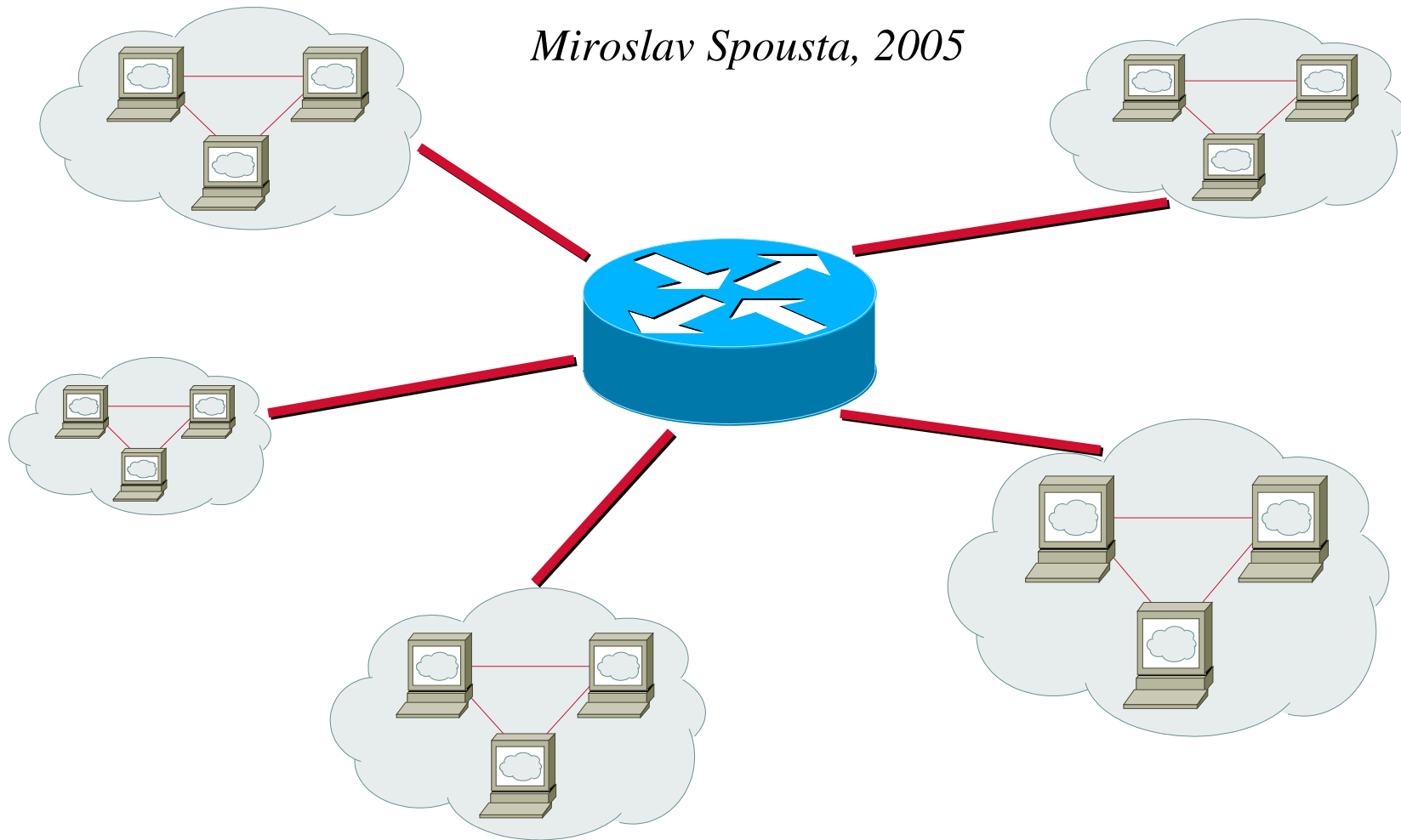


# Počítačové sítě II

## 16. elektronická pošta

*Miroslav Spousta, 2005*



# Elektronická pošta

- služba, která slouží kvýměně zpráv
- existuje mnoho standardů (firemních i veřejných)
- Internetová pošta (SMTP)
  - dneska asi nejrozšířenější
- MS Mail (Microsoft)
- X.400 (telekomunikační standard)
  - komplikované adresy typu G=Petr;S=Novak;O=cuni;OU=rektorat,C=cz
- přenos pomocí UUCP (Unix to Unix CoPy)
  - přenos souborů, zpráv před rozšířením Internetu
  - adresování pomocí vytyčení cesty k cíli přes propojené servery (hop)
  - např.: !{bighost,mail}!alpha!beta!novak
- nejsou vzájemně kompatibilní

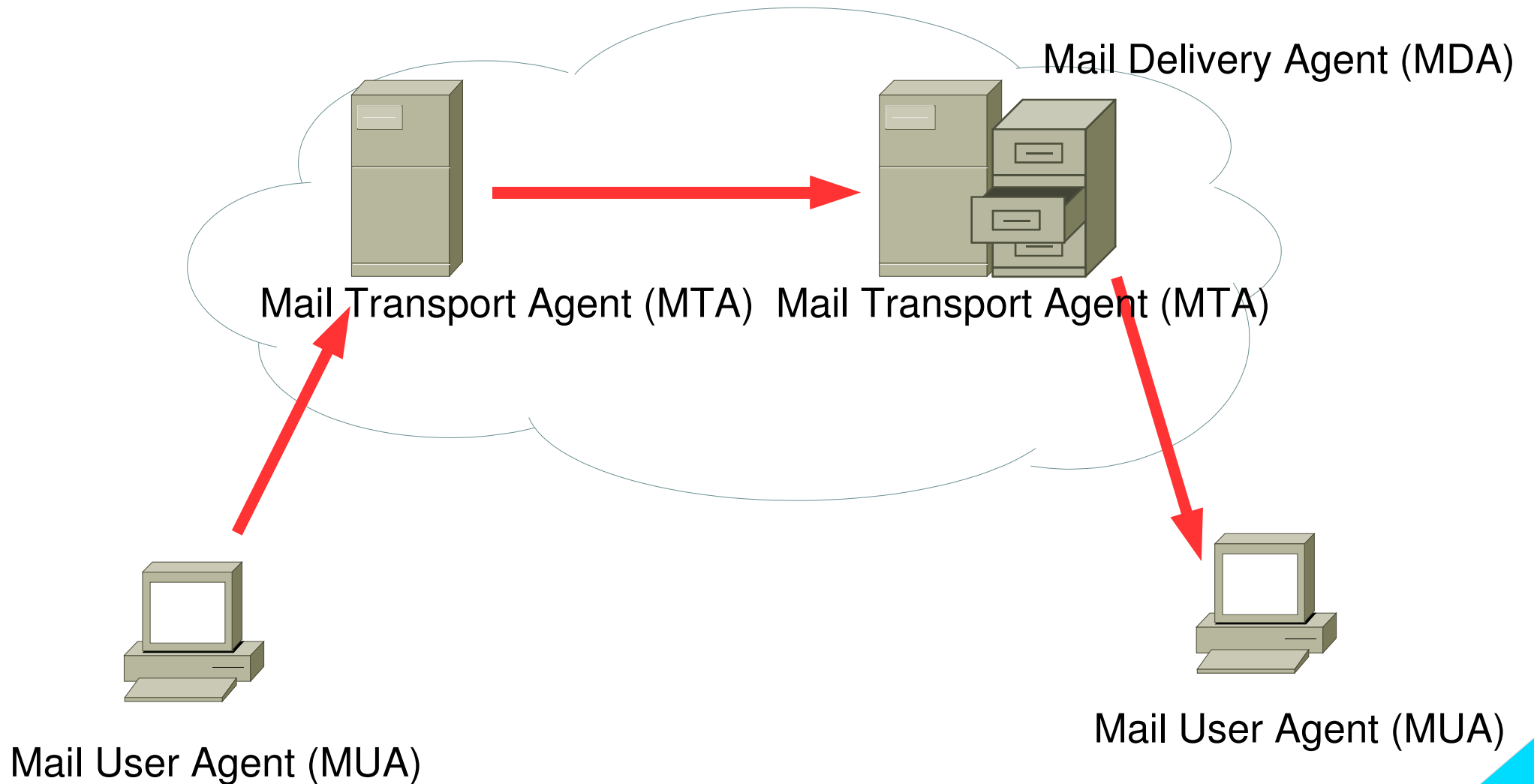
# Elektronická pošta

- co musí definovat standard pro elektronickou poštu:
- formát zpráv
  - jak se bude zpráva dělit. kolik může mít částí
  - které údaje jsou povinné a které volitelné
- formát adres
  - v jakém formátu se bude zapisovat odesílatel a příjemce
- protokol pro přenos pošty mezi servery
  - jak si servery budou vyměňovat zprávy
- protokol pro poslání zprávy
- protokol pro získání zprávy
  - jak se klient dostane ke zprávám

# Terminologie

- MUA = Mail User Agent
  - program, který běží na počítači uživatele
  - slouží pro interakci s uživatelem (psaní a čtení zpráv)
  - např. MS Outlook, Mozilla Thunderbird, The Bat!, ...
- MTA = Mail Transport Agent
  - je zodpovědný za doručování pošty od odesílatele k příjemci
  - většinou servery elektronické pošty po cestě
  - např. sendmail, Postfix, Exim, MS Exchange
- MDA = Mail Delivery Agent
  - program, který poštu doručuje do schránky na cílovém serveru
  - např. procmail, maildrop, lmtpl (Unix) nebo součástí MTA (MS Exchange)
- poštovní schránka: pro každého uživatele, doručuje se do ní pošta

# Terminologie



# Elektronická pošta v Internetu

- v Internetu: SMTP (Simple Mail Transport Protocol)
  - použitelný i mimo Internet
  - používá spolehlivou službu (v Internetu TCP)
  - vznikl původně jako jednoduchý protokol pro přenos zpráv (co nejpodobnější reálnému světu)
  - RFC822 (RFC2822)
- SMTP služba je rychlá
  - doručování – sekundy až minuty
- SMTP služba je spolehlivá
  - komunikace je navrhovaná tak, aby nedošlo ke ztrátě zpráv při neočekávaných situacích
  - je dané, kdo je zodpovědný v každé chvíli za danou zprávu
- SMTP je efektivní
  - jednoduchý formát, umožňuje automatické hromadné zpracování

# SMTP pošta

- původně pouze pro přenos textových (ASCII) zpráv
  - dnes i přílohy, národní znaky, zprávy skládající se z několika částí
- je jednoduchý (textový)
  - srozumitelný i člověku (testování)
  - ale dobře zpracovatelný automaticky
- funguje off-line
  - příjemce a odesílatel nemusí být ve stálém připojení
  - odesílatel zprávu pošle, ta se zařadí do fronty a počká, až ji bude možné doručit
  - příjemce vyzvedává svojí zprávu také nezávisle
- původně uživatel přistupoval k poště na stejném počítači, jako má umístěnu poštovní schránku
  - dneska většinou vzdálený přístup, resp. rozdělená poštovní schránka

# RFC

- RFC 822 (RFC 2822) definuje
  - formát zprávy
    - jak vypadá hlavička zprávy, z čeho se skládá
    - které položky jsou povinné, které volitelné)
  - tělo zprávy
    - v jakém je formátu, jak je odděleno od hlavičky
- RFC 821 (RFC 2821) definuje
  - protokol pro přenos pošty mezi MTA: SMTP
    - zahájení a ukončení přenosu a doplňující příkazy
- RFC 2045 – 2049 (MIME)
  - rozšiřují možnosti pošty o přílohy
    - strukturování těla zprávy a ukládání binárních dat
  - umožňují používat národní jazyky v hlavičkách



# Formát zprávy

- zprávy jsou kódovány jako text (v US-ASCII)
  - konce řádků jsou Internetové: CRLF
  - řádky mají maximální velikost 998 znaků (+2 CR a LF)
- zpráva se skládá z hlavičky a těla zprávy
  - tyto dvě části jsou odděleny prázdným řádkem
- položky hlavičky se skládají z jména položky, které následuje dvojtečka (bez mezery, např. Subject:) a za ním následuje obsah položky
  - obsah některých položek má pevný formát (adresy, datum, ...), tzv. strukturované položky
  - jiné položky mají volný formát, tzv. nestrukturované položky
  - položky mohou být rozděleny na několik řádek, pak pokračující řádky musí začínat bílým znakem (mezera, tabelátor, ...)
  - na pořadí položek *nezáleží*
- tělo obsahuje řádky textu v US-ASCII

# Formát zprávy

Received: from SKOPALOVA (mx.vsfs.cz [213.210.148.2]  
by smtp.nextra.cz (Postfix) with ESMTTP id 92EBE5DA0  
for <qiq@ucw.cz>; Tue, 12 Apr 2005 09:53:46 +0200  
(CEST)

From: Hana Skopalova <[hana.skopalova@vsfs.cz](mailto:hana.skopalova@vsfs.cz)>

To: Miroslav Spousta <[qiq@ucw.cz](mailto:qiq@ucw.cz)>

Subject: Vyuka

Date: Tue, 12 Apr 2005 09:58:40 +0200

Dobry den, nezapomente na vyuku!

# (některé) položky SMTP hlavičky

- **From:**
  - adresa odesílatele (člověk, proces, ...), povinná položka
- **Sender:**
  - skutečný odesílatel zprávy (např. sekretářka)
- **Reply-To:**
  - adresa, na které se očekává odpověď, používá se např. u konferencí
- **To: Cc:, Bcc:**
  - příjemce zprávy, příjemce kopie, příjemce slepé kopie (ostatní nevidí)
  - povinná je aspoň jedna z těchto tří položek
- **Date: nebo Resent-Date:**
  - čas odeslání (přeoslání) zprávy, formát: **Tue, 19 Apr 2005 18:37:52 +0200**
  - povinná položka

# (některé) položky SMTP hlavičky

- **Received:**

- cesta, kudy e-mail putoval internetem
- každý MTA po cestě přidá na začátek zprávy tuto položku
- nesmí měnit obsah předcházejících položek
- posloupnost umožňuje vystopovat, kudy zpráva prošla (a jak)
  - první věc, na kterou se zaměřit při diagnostice problémů
- má mnoho volitelných položek: from (odkud), by (kým), via (fyzická cesta), with (protokol), id (identifikace u příjemce), for (obálková adresa)
- jedna povinná položka: čas a datum

- **Return-Path:**

- kam se posílá zpráva zpět jako nedoručitelná

- **Subject:**

- stručný obsah zprávy

# (některé) položky SMTP hlavičky

- **Message-Id:**

- identifikace zprávy
- měla by být unikátní v Internetu, přesný formát není definovaný
- dá se podle ní identifikovat, zda se jedná o tutéž zprávu, nebo ne
- hodí se např. pro detekci smyček

- **X- :**

- speciální (rozšiřující) hlavičky, jsou ignorovány
- např.: X-Status, X-Mailer, X-Spam-Status, ...

- **Status:** a další nestandardní hlavičky

- přidává např. MUA pro zapamatování, jestli zpráva byla přečtena, nebo ne

# Formát adresy

- dřívější formát: *login@host.domena*
  - např. *qiq@jabberewock.ucw.cz*
  - adresa je vázána na počítač
  - málo pružné (co když přijde nový server?)
- dnes se používá: *jmeno@domena*
  - např. *qiq@ucw.cz*
  - z DNS se zjistí, na který stroj se doručuje pošta pro doménu ucw.cz
  - může jich být víc
- formát zápisu adresy dle RFC 822:

*Identifikace <jmeno@domena>*

*jmeno@domena (Identifikace)*

*jmeno@domena*

# Poznámky k adresám

- doménová část adresy není case-sensitive (DNS)
- to, co je před znakem „,@“ může a nemusí být case-sensitive
  - záleží na implementaci – MUA musí počítat s tím, že na velikosti písmen záleží
- URL je ve formátu <mailto:qiq@ucw.cz>
- adresa [Postmaster@domena](mailto:Postmaster@domena) by měla být vždy platná a měl by ji číst správce daného poštovního serveru
- hlavička Bcc se ze zprávy před odesláním odstraní
  - adresáti, kteří v ní byli uvedeni dostanou kopii takovéto zprávy (spolu s příjemci v Cc: a To:)

# Doručení pošty

- zpráva je ze stanice odesilatele předána pomocí protokolu SMTP serveru (MTA) ke zpracování
- MTA zprávu přijme a převezme za ni zodpovědnost
  - většinou ji uloží na disk
  - má za úkol zprávu doručit na cílový mail server, případně ji vrátit zpět jako nedoručitelnou
- MTA zjistí z DNS poštovní servery (může jich být víc) pro danou doménu (MX záznam v DNS pro danou doménu)
  - neexistuje-li v DNS MX záznam pro danou doménu, zkusí najít A záznam a na tuto adresu zprávu doručit
  - MX záznamy mají přiřazenou prioritu – nejnižší číslo znamená nejvyšší prioritu
- MTA se pokusí doručit zprávu na cílový server (servery) podle priority
  - pokud se mu to nevede, zpráva zůstává ve frontě na daném MTA



# Doručení pošty 2

- cílový MTA zprávu předá MDA k uložení do uživatelské schránky
  - odtud si ji uživatel pomocí MUA může vyzvednout
  - případně pomocí protokolů POP3 nebo IMAP
- pokud se MTA nedaří doručení po určitou dobu (typicky 5 hodin), pošle odesilateli (Return-Path:) upozornění, zprávu si nechává ve frontě
- pokud se MTA nedaří doručení po dlouhou dobu (3 dny), vrátí zprávu jako nedoručitelnou
- obecně se předpokládá, že poštovní servery budou mít permanentní připojení k Internetu
  - pokud některý server není přímo připojen, měl by existovat jiný poštovní server pro danou doménu, který bude mít nižší prioritu
  - doručení zpráv proběhne po připojení „primárního“ mail serveru

# SMTP

- Simple Mail Transfer Protocol, RFC 821 (2821)
- definuje, jak vypadá komunikace mezi MTA
  - a mezi MUA a MTA při poslání zprávy
- přenáší RFC 822 zprávy
- vychází z původních požadavků
  - jednoduchost, efektivita: textový protokol
  - snadná rozšiřitelnost: ESTMP
  - přenos US-ASCII znaků: 7bitový přenos
- rozdělení (analogie klasické pošty):
  - list papíru: zpráva (hlavička – jako na hlavičkovém papíru)
  - obálka: součást SMTP protokolu (některé položky se zapisují i do zprávy)
- SMTP přenáší zprávy podle obálek, ne podle obsahu (listu papíru)

# Obálka a zpráva

From: <qiq@ucw.cz>

RFC 822

To: <test@mail.vsfs.cz>

Subject: test testov testovic

ddd test

RFC 821

MAIL FROM: <qiq@seznam.cz>

RCPT TO: <test@mail.vsfs.cz>

# SMTP protokol

- spojení probíhá na portu 25 (587)
  - na IP adresu MX nebo A DNS záznamu pro danou doménu
- nejprve se servery vzájemně pozdraví příkazem HELO
- poté odesílající server předá cílovému serveru údaje z hlavičky
- odesílatel: MAIL FROM: <qiq@ucw.cz>
- příjemce: RCPT TO: <spousta@mail.vsfs.cz>
- následuje příkaz DATA, po kterém se pošle celá zpráva (hlavičky následované prázdným řádkem a tělem zprávy)
- nakonec se spojení ukončí: QUIT
- v rámci jednoho spojení je možné poslat několik zpráv (HELO se zadává pouze napoprvé, pak už jen MAIL FROM, RCPT TO, DATA)

# SMTP konverzace

```
S: 220 www.example.com ESMTP Postfix
C: HELO mydomain.com
S: 250 Hello mydomain.com
C: MAIL FROM: <sender@mydomain.com>
S: 250 Ok
C: RCPT TO: <friend@example.com>
S: 250 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: Subject: test message
C: From: sender@mydomain.com
```

```
C: To: friend@example.com
C:
C: Hello,
C: This is a test.
C: Goodbye.
C: .
S: 250 Ok: queued as 12345
C: quit
S: 221 Bye
```

# SMTP: chybové kódy

- chyby a stavy v číselné a textové podobě
  - text je určen pro administratory/uživatele
  - číslo udává, co nastalo za chybu
- číslo je tříciferné, každá cifra udává jiný typ zprávy
  - první cifra: úspěch (1-3), chyba (5), dočasná chyba (4)
  - druhá cifra: kategorie chyby: syntax (0), spojení (2), zpráva (5)
  - třetí cifra: konkrétní chyba v dané kategorii

# SMTP: chybové kódy

211 System status, or system help reply

214 Help message

220 <domain> Service ready

221 <domain> Service closing transmission channel

250 Requested mail action okay, completed

251 User not local; will forward to <forward-path>

252 Cannot VRFY user, but will accept message and attempt delivery

354 Start mail input; end with <CRLF>.<CRLF>

421 <domain> Service not available, closing transmission channel

450 Requested mail action not taken: mailbox unavailable (e.g., mailbox busy)

451 Requested action aborted: local error in processing

452 Requested action not taken: insufficient system storage

# SMTP: chybové kódy

- 500 Syntax error, command unrecognized
- 501 Syntax error in parameters or arguments
- 502 Command not implemented (see section 4.2.4)
- 503 Bad sequence of commands
- 504 Command parameter not implemented
- 550 Requested action not taken: mailbox unavailable
- 551 User not local; please try <forward-path>
- 552 Requested mail action aborted: exceeded storage allocation
- 553 Requested action not taken: mailbox name not allowed
- 554 Transaction failed



# SMTP: poznámky

- adresy uvedené na obálce (v SMTP MAIL FROM: a RCPT TO:) se používají na opravdové doručení zprávy – MTA nehledí na údaje, které jsou uvedeny v hlavičkách (!!)
- Bcc: se řeší tak, že na obálce je skutečný příjemce a zpráva je shodná se všemi ostatními příjemci
  - neboli ze zprávy MUA odstraní Bcc: hlavičku a stejnou zprávu pošle na všechny adresy v To:, Cc: a Bcc:
- RCPT TO: se může v SMTP dialogu opakovat (šetříme pásmo – zpráva se přenáší po lince pouze jednou)
- [Postmaster@domena](#) je vždy platná adresa

# SMTP: další příkazy

- VRFY
  - ověří, že adresát existuje
  - dnes se kvůli spammerům/hackerům zakazuje
- EXPN
  - zobrazí obsah distribučního listu (seznam adres)
  - platí o něm to samé, co o VRFY
- RSET
  - zrušení přenosu zprávy
- NOOP
  - prázdná operace
- HELP
- QUIT

# ESMTP

- extended SMTP (RFC 1651)
  - rozšíření SMTP o další možnosti
  - místo HELO na počátku konverzace se použije příkaz EHLO
  - pokud projde, MTA podporuje ESMTP, vypíše podporovaná rozšíření
- doručení (pozitivní i negativní): DSN
  - delivery status notification
  - RFC 1891
- maximální velikost mailu: SIZE
  - umožňuje serveru odmítnout příliš velkou zprávu ještě před začátkem přenosu
- pipeline režim: PIPELINING
  - umožňuje vykonávat více příkazů bez čekání na odpověď server
- osmibitový přenos (není potřeba speciální kódování pro MIME): 8BITMIME

# MIME

- Multipurpose Internet Mail Extensions, RFC 2045 – 2049
  - používá se nejen pro poštu, ale je součástí i např. HTTP
- mechanismus, jak přenášet pomocí SMTP libovolné zprávy
  - strukturované, binární, ...
- zpětně kompatibilní
  - dnes všechny MUA podporují MIME
- SMTP přenáší data sedmibitově
  - nejvyšší bit nemusí být přenesen
  - a omezuje maximální délku řádku na 998
- MIME přidává k datům jejich popis (typ)
- používá speciální kódování (BASE64 a quoted printable), aby přeneslo binární (osmibitová) data přes 7bitový kanál
- definuje také kódování dat v položkách hlavičky

# MIME typ

- RFC 2046
- MIME typ se skládá ze dvou částí: typu (obecného) a podtypu (konkrétní)
  - odděleny jsou lomítkem
  - registruje IANA
- obecný typ: text, application, image, audio, video, message, multipart, model
- text/plain, text/html, text/rtf...
  - čistě textové části
  - může u nich být uvedeno kódování (text/html; charset=iso-8859-2)
- application/postscript, application/msword, application/octet-stream
  - binární data, spustitelné soubory
- image/jpeg, image/png, image/gif, image/tiff, ...
  - obrázky

# MIME typ 2

- audio/mpeg, ...
  - audio soubory
- video/mpeg, video/quicktime, ...
  - video soubory
- message/rfc822
  - vložená zpráva (podle RFC (2)822)
    - tedy hlavičky a tělo zprávy
  - např. chybové hlášení, odpověď s vloženým původním dopisem, ...
- message/partial
  - část zprávy
  - používá se v případě, že chceme odeslat velkou zprávu (kterou servery po cestě nepodporují)
  - klient sestaví části do původní zprávy

# MIME typ multipart

- multipart/mixed
  - různé typy příloh (částí)
- multipart/alternative
  - části jsou vzájemně zástupné – MUA ukáže tu, kterou umí zobrazit nejlépe
  - např. text/plain a text/html
- musí obsahovat atribut (parametr boundary)
  - udává, kde začínají jednotlivé části
  - oddělovač je uvozen `--`, musí být na začátku řádku
  - poslední oddělovač je zakončen také pomocí `--`
  - každá část se skládá z hlavičky, prázdného řádku a těla (vlastně jako rfc822 zpráva)
    - v této hlavičce jsou pouze MIME položky
  - pokud není uveden typ (Content-type), použije se text/plain; charset=US-ASCII

# MIME

```
Fom: Denis Vlasenko <vda@port.imtp.ilyichevsk.odessa.ua>
To: linux-kernel@vger.kernel.org
MIME-Version: 1.0
Content-Type: Multipart/Mixed; boundary="Boundary-00=_uNKZC5QbjnCd98x"

--Boundary-00=_uNKZC5QbjnCd98x
Content-Type: text/plain; charset="koi8-r"
Content-Transfer-Encoding: 7bit
Content-Disposition: inline
This + next patch were "modprobe tcrypt" tested.

--Boundary-00=_uNKZC5QbjnCd98x
Content-Type: text/x-diff; charset="koi8-r"; name="1.be.patch"
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="1.be.patch"
...
--Boundary-00=_uNKZC5QbjnCd98x--
```



# MIME: položky hlavičky

- informace o MIME se ukládají v hlavičkách zprávy
- zprávy mohou být rozčleněny do několika částí (stromová struktura)
  - pomocí typu multipart
- povinná položka hlavičky: **MIME-Version**
  - v současné době 1.0
- volitelné položky: **Content-type**, **Content-transfer-encoding**, **Content-id**, **Content-description**, **Content-disposition**
- **Content-type**
  - udává, jakého typu je daná zpráva (MIME type)
  - např. prostý text v ASCII: text/plain; charset=US-ASCII
- **Content-transfer-encoding**
  - jaké se použilo kódování pro přenos: např. quoted-printable, base64, 7bit, 8bit

# BASE64

- způsob kódování 8bitových znaků pomocí šesti bitů (3x8b -> 4x6b)
- vstupní znaky (bitová reprezentace) se rozdělí na šestice bitů, a ta se zakóduje pomocí malých a velkých písmen, čísel a znaků „/“ a „+“
- vzájemně jednoznačné přiřazení (je potřeba dekodovat :-))
- hodí se pro binární data, formátuje se na řádky dlouhé 72 znaků

P ř í š e r k á m

01010000	11111000	11101101	10111001	01100101	01110010	01101101	11100001	01101101			
010100	001111	100011	101101	101110	010110	010101	110010	011011	011110	000101	101101

U P j t u W V y a + F t

0	1	2	3	...	25	26	27	28	...	51	52	53	...	61	62	63
A	B	C	D	...	Z	a	b	c	...	z	0	1	...	9	+	/

# Quoted printable

- znaky, které jsou součástí US-ASCII necháme tak, jak byly
- ostatní znaky zakódujeme sekvencí „=“ následované hexadecimálním vyjádření znaku
  - např. v iso-8859-2 má „ř“ hexadecimální kód 0xF8, v quoted printable tedy „ř“ bude =F8
  - platí i pro „=“, ...
- výhodné pro texty, ve kterých je poměrně málo znaků mimo US-ASCII
- texty zakódované pomocí quoted printable mají řádky dlouhé maximálně 76 znaků (pokud jsou delší, rozdělí se)

Příšerkám => P=F8=ED=B9erk=E1m

# MIME: další položky hlavičky

- **Content-id**

- identifikace části zprávy
- např. pro multipart/alternative zprávy vyjadřuje, které zprávy jsou přímo zastupitelné (stejně id)

- **Content-description**

- popisuje obsah, nezpracovává se, pouze se zobrazí uživateli

- **Content-disposition**

- **inline** – zobrazit jako součást zprávy
- **attachment** – zobrazit zvlášť jako přílohu, může mít atribut filename se jménem souboru, který je přiložen (při uložení přílohy se vytvoří soubor s tímto jménem)

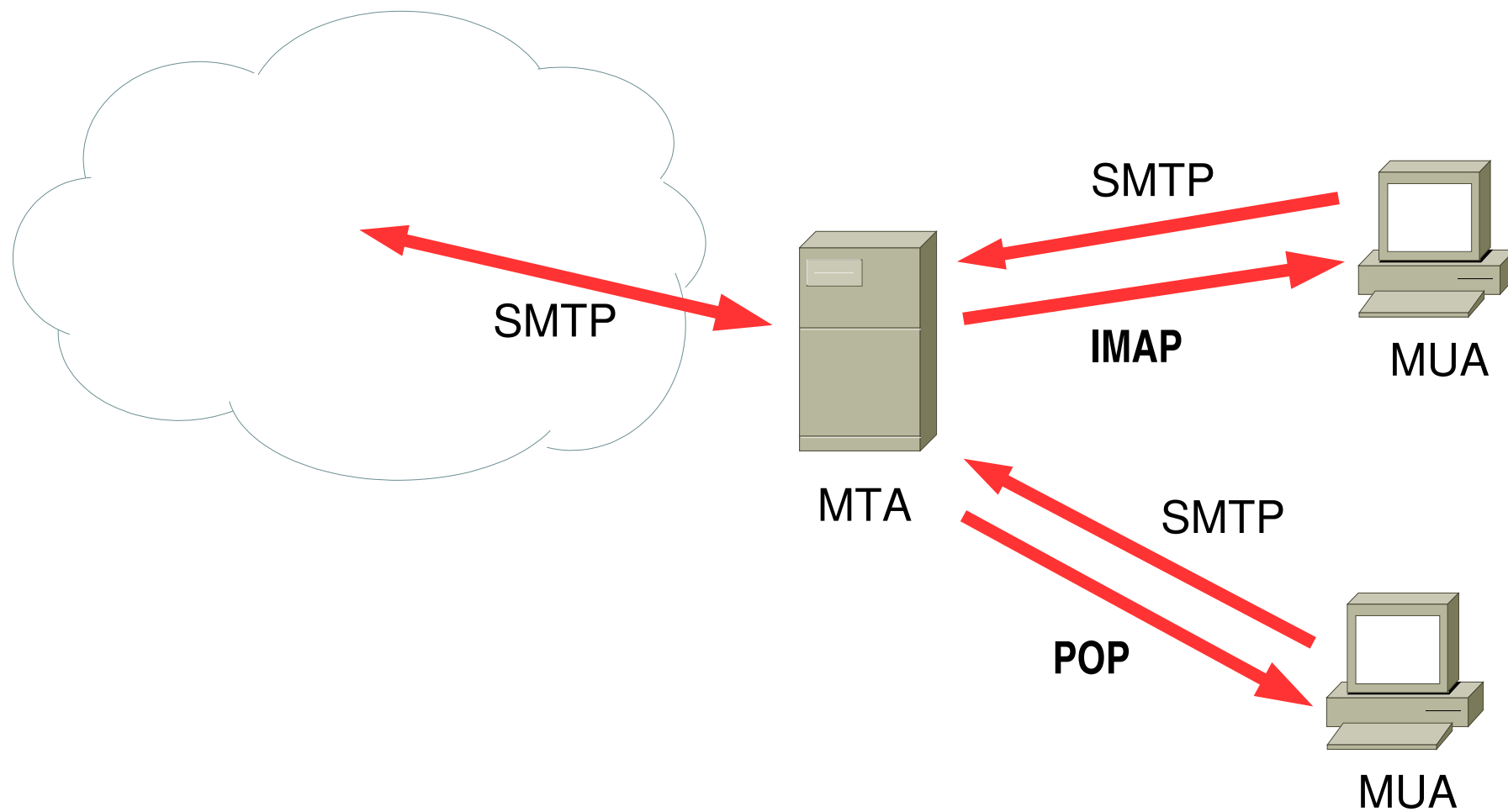
# MIME: hlavičky

- kromě těla zprávy je potřeba přenášet národní znaky i v položkách hlavičky
  - Subject, From, To, ...
- MIME umožňuje v položce hlavičky uvést řetězec speciální řetězec ve formátu: `=?xxx?yyy?zzz?='`
  - xxx je použitá znaková sada (např. iso-8859-2)
  - yyy je použité kódování (znak B pro BASE64, Q pro upravené quoted printable)
    - u Q může být mezera nahrazena znakem „\_“, nebo =20, bílé znaky jsou kódovány, ...
  - zzz je zakódovaná zpráva
- příklady:
  - From: `=?iso-8859-2?Q?Hana_Skopalov=E1?=' <hana.skopalova@vsfs.cz>`
  - Subject: `=?iso-8859-2?B?UmU6ILlrb2xu6Q==?='`

# Vyzvednutí pošty

- MDA doručí zprávu do schránky adresáta
  - do souboru, databáze, ...
  - jak se k ní klient (MUA) dostane?
  - MTA by měl být trvale připojen k Internetu, MUA být nemusí
- přímý přístup (mbox, maildir, databáze, ...)
  - MUA běží na stejném počítači, nebo je schránka exportovaná přes FS
  - klient musí rozumět formátu schránky
  - je potřeba zamýkat schránku (přistupuje současně MUA a MDA)
- vzdálený přístup
  - protokoly POP a IMAP (pouze vyzvednutí, posílání – SMTP(!))
  - umožňují přistupovat ke schránce z libovolného počítače v Internetu
  - standard, RFC
  - liší se ve filosofii a možnostech

# Vyzvednutí pošty



# POP3

- Post Office Protocol verze 3, RFC 1939
- používá TCP, port 110
- protokol je textový (podobně jako SMTP), velmi jednoduchý
- umožňuje stáhnout zprávy ze schránky na klientský počítač (do MUA)
  - neboli zprávy se doručí na klientský počítač
- hodí se pro *off-line* MUA, který se připojuje k serveru pouze na omezenou dobu (pro přenos zpráv)
- umožňuje přistupovat pouze do jedné schránky (INBOX)
- některé servery umožní ponechání zpráv na serveru
  - protokol s tím původně nepočítal
  - případně může být nastaveno mazání přečtených zpráv po uplynutí nějaké doby



# POP3 protokol

- příkazy jsou maximálně čtyřpísmenné, argumenty jsou odděleny mezerou
- server vrací odpověď „+OK“, nebo „-ERR“ (následovat může chybová hláška)
- spojení má tři fáze: autentizační (přihlášení uživatele) a transakční (stahování, mazání pošty), aktualizací (opravdové smazání pošty)
- příkazy:
  - USER string (login), PASS string (heslo)
  - LIST (výpis čísel zpráv), RETR n (stažení zprávy) DELE n (smazání zprávy)
  - QUIT (ukončení), RSET („obnovení“ smazaných zpráv)
- existují rozšíření (podobně jako v SMTP)
  - zjišťují se příkazem CAPA
  - např. maximální doba, po kterou může být přečtená zpráva na serveru (EXPIRE)
  - stažení jen části zprávy (např. pouze hlavičky): TOP, unikátní čísla zpráv: UIDL

# POP3 protokol

```
S: +OK mail Cyrus POP3 v2.1.18 server
  ready <2330620496.1114241905@mail>
C: USER test
S: +OK Name is a valid mailbox
C: PASS testpass
S: +OK Maildrop locked and ready
C: LIST
S: 1 7609
S: 2 7684
S: 3 7440
S: 4 6965
S: .
C: RETR 1
S: From: xxx@yyy.cz
S: Subject:...
```

```
S:
S: Dobry den!
S: Tesilo me
S: .
C: DELE 1
S: +OK message deleted
C: QUIT
S: +OK
```

# IMAP

- Internet Message Access Protocol (verze 4rev1), RFC 3501
- používá TCP, port 143
- původně pouze pro on-line přístup (nižší verze protokolu)
- nyní umožňuje i off-line práci díky cachování zpráv na klientovi
  - dokonce je možné zprávy mazat s tím, že synchronizace se provede později
- zprávy jsou uloženy na serveru
  - na klientovi je pouze kopie zpráv pro rychlejší načítání (cache)
  - jsou stále přístupné, z různých míst v Internetu (např. z práce a z domova (i současně!))
- umožňuje vytvářet hierarchii složek
  - stromová struktura, jména v UTF-7
- umožňuje nastavovat atributy zprávám, přesouvat zprávy mezi složkami
- mnoho různých způsobů autentizace

# IMAP protokol

- spojení má čtyři stavy (fáze): neautentizovaný, autentizovaný, vybraný mailbox, logout
- každý příkaz klienta je uvozen identifikací (tag) operace
  - aby bylo možné identifikovat odpověď – je možné paralelně provádět několik operací
- umožňuje přihlašovat a odhlašovat složky
  - většinou se na příchozí poštu kontrolují pouze přihlášené složky
- každá zpráva by měla mít unikátní číslo
- čísla v mailboxech jsou rostoucí (definují pořadí)
- protokol umožňuje vyhledávání na serveru

# IMAP protokol

```
S: * OK IMAP4rev1 Service Ready
C: a001 login mrc secret
S: a001 OK LOGIN completed
C: a002 select inbox
S: * 18 EXISTS
S: * FLAGS (\Flagged \Deleted \Seen)
S: * 2 RECENT
S: * OK [UNSEEN 17] Message 17 is the
first unseen message
S: * OK [UIDVALIDITY 385752] UIDs valid
S: a002 OK [READ-WRITE] SELECT completed
S: a003 OK FETCH completed
C: a004 fetch 12 body[header]
S: * 12 FETCH (BODY[HEADER] {342}
S: Date: Wed, 17 Jul 1996 02:23:25 -0700
S: From: Gray <gray@cac.washington.edu>
```

```
S: Subject: IMAP4rev1 WG mtg summary
S: To: imap@cac.washington.edu
S: cc: minutes@CNRI.Reston.VA.US
S: Message-Id: <B27397-
0100000@cac.washington.edu>
S: MIME-Version: 1.0
S: Content-Type: TEXT/PLAIN; CHARSET=US-
ASCII
S:
S: )
S: a004 OK FETCH completed
C: a006 logout
S: * BYE IMAP4rev1 server terminating
connection
S: a006 OK LOGOUT completed
```

# spam

- nevyžádaná sdělení
  - šířená telegramem, telefonem, e-mailem, ...
  - původně z Monty Python
  - později „shit posing as spam“
- komerční: Usenet, 1994
- analogie podomního prodeje
  - dneska letáky ve schránce (junk mail)
- s rozvojem elektronických komunikací nastal jeho masivní rozvoj
  - snadné, levné, masové, účinné
  - je těžké se bránit
- mnoho podob: e-mail, IM, blog, diskuse, spamdexing



# e-mail SPAM

- nejrozšířenější forma spamu
- rozesílán na mnoho adres
  - získaných z různých služeb vyžadujících registraci
  - diskusních skupin
  - webových stránek
- rozšířily se antispamové filtry
  - roztočila se spirála zdokonalování spamů a antispamů
  - udává se, že kolem 80% dnešních e-mailů jsou spamy
- zatěžují mailové servery
- znesnadňují komunikaci
  - především antispamy
- spamerů není mnoho, ale dokáží znepríjemnit život stamiliónům uživatelů

# Triky spamerů

- na počátku obyčejné e-mailu s reklamní tematikou
  - filtrování podle hlaviček/údajů na obálce
  - položky v hlavičce lze snadno měnit (stejně jako u klasické pošty (snail mail))
  - dnes je adresa odesilatele buď neplatná, nebo kradená (a e-mail obsahuje odkaz na web),
  - neboli podle hlavičky e-mailu toho moc nezjistíme
- rozpoznání spamu podle obsahu
  - nejčastěji se jedná o reklamu na Viagra, zvětšení přirození, ...
  - je možné používat stop slova (pokud se v e-mailu najde slovo Viagra, zahodíme ho)
  - není to vhodné (může se objevit i v korektním e-mailu)
  - realce spamerů: V1ágra



# Triky spamerů

- podstata spamování: informace je rozesílána na obrovské množství adres
  - nápad: pokud jeden uživatel označí zprávu jako spam, ostatní už to budou vědět
  - měli by vždy označovat lidé
  - kolaborativní sítě (razor, pyzor)
  - pozor na zneužití
  - porovnávají se pouze hashe zpráv
  - reakce spamerů: přidávají do zpráv náhodnou část, pro každý spam jinou
  - obrana (částečná): hash počítáme z náhodných částí spamu
- učící se statistický filtr
  - idea: pro každý e-mail spočítáme pravděpodobnost, že se jedná o spam
  - filtr se musí nejprve „naučit“ informace z předhozených spamů a nespamů (hamů)

# Bayesovský filtr

- idea: pro každý e-mail spočítáme pravděpodobnost, že se jedná o spam
- pro každé slovo si pamatuje, jaká je pravděpodobnost, že e-mail obsahující toto slovo je spam
  - na počátku vezmeme dvě složky, jedna se spamem a druhá s hamem
  - pro každé slovo spočítáme  $P(w) = \text{počet výskytů } w \text{ (spam)}/\text{počet všech výskytů } w$
  - tím získáme pravděpodobnost, že dané slovo je součástí spamu
  - zde je potřeba učení
- při hodnocení e-mailu se spočítá geometrický průměr z pravděpodobností, že slova v e-mailu určují spam
  - tento postup se kvůli optimalizacím aplikuje pouze na zajímavá slova (ta, která jsou extrémní – buď indikují, že se jedná o spam, nebo ta, která indikují ham)
- filtr je potřeba naučit pomocí hamů pro každého uživatele (skupinu uživatelů zvlášť)

# Další triky spamerů

- jak zmást bayesovský filtr
  - vkládat nesmyslná slova (budou vytvářet dojem, že se jedná o ham)
    - dneska se používá naopak pro detekci
  - vkládat odstavce z knih, obsahem sdělení je třeba jen obrázek, zbytek se nezobrazí
    - filtrování podle obsahu nefunguje, je potřeba, aby nastoupilo jiné
- filtrování podle hlaviček Received:
  - ty jsou (od jistého bodu) korektní i u spamu
  - existují databáze serverů, ze kterých pochází spam
  - jsou řešené přes DNS (do DNS se položí dotaz, pokud server vrátí odpověď (server zná), jedná se o server posílající spamy
  - spameři začali používat cizí počítače pro rozesílání
    - často jsou to počítače unesené (napadené), nebo tzv. open relay
- open relay: server, který umožňuje každému přeposlat e-mail

# Další triky spamerů

- jak zabránit falšování adres v e-mailu?
  - původní RFC s tím nepočítala
  - iniciativa SPF – sender policy framework (<http://spf.pobox.com>)
  - řeší posílání e-mailu s falešnou adresou odesilatele
  - do DNS pro doménu přidává záznam, které servery jsou platné pro odchozí poštu pro danou doménu
  - je snadné zkontrolovat, zda pošta přichází ze serveru, který je platný odchozí server
  - zatím se málo používá (ale např. AOL, gmail to používají)
- poslat korektně se tvářící e-mail, který obsahuje pouze odkaz na web, kde je reklama
- dnes existují i databáze takovýchto serverů (spamer si jich nemůže registrovat příliš mnoho – stojí to peníze)
- jako ochrana je také používáno striktní vyžadování RFC

# Ochrana proti spamům

- většinou pomocí vyhodnocení obsahu a dalších faktorů
- používá se kombinace několika metod
- každému e-mailu se přiřadí skóre, které určuje, jak moc je to spam
- uživatel si zvolí, kde je hranice
- jedná se o neutuchající boj, podobně, jako v případě virů/červů
- ochrana může být nasazena na MTA (MDA) a/nebo na MUA
  - často společně s kontrolou virů a červů